Final Project

Christophe Hunt

May 21, 2017

Contents

1 Variable

Pick one of the quantitative independent variables from the training data set (train.csv), and define that variable as X.

Pick SalePrice as the dependent variable, and define it as Y for the next analysis.

1.1 Variable Picked

> The variable we will set to X is LotArea, which is defined as the Lot size in square feet. I chose LotArea because an anecdotal assumption is that the larger the lot size is the higher the sale price. However, living in NYC, I know that tiny lots in very desirable places have sold for a high price so I believe there may be some interesting varability.

```
library(tidyverse)
train.df  <- as_tibble(read.csv(paste("https://raw.githubusercontent.com/",
                                      "ChristopheHunt/",
                                      "MSDA---Coursework/master",
                                      "/Data%20605/Final%20Project/train.csv",
                                      sep = "")))
```

```
sub.train.df <- train.df[, c("SalePrice", "LotArea")]
```

2 Probability

Calculate as a minimum the below probabilities a through c.

Assume the small letter “x” is estimated as the 4th quartile of the X variable, and the small letter “y” is estimated as the 2nd quartile of the Y variable. Interpret the meaning of all probabilities.

```
prob.x <- list(qrt = as.numeric(quantile(sub.train.df$LotArea)[4]))

prob.y <- list(qrt = as.numeric(quantile(sub.train.df$SalePrice)[2]))
```

```
prob.y.x <- sub.train.df %>% mutate(greaterLotArea = ifelse(LotArea >= prob.x$qrt, 1, 0),
                                     lesserLotArea = ifelse(LotArea < prob.x$qrt, 1, 0),
                                     greaterSalePrice = ifelse(SalePrice >= prob.y$qrt,1, 0),
                                     lesserSalePrice = ifelse(SalePrice < prob.y$qrt,1, 0))
```

2.1 a. $P(X > x|Y > y)$

```
a <- (sum(ifelse(prob.y.x$greaterLotArea == 1 &
                   prob.y.x$greaterSalePrice == 1, 1, 0))
      / nrow(prob.y.x)) / ((sum(prob.y.x$greaterLotArea) / nrow(prob.y.x)))
a
```

```
## [1] 0.9369863
```

2.2 b. $P(X > x, Y > y)$

```
b <- sum(ifelse(prob.y.x$greaterLotArea == 1 &
                  prob.y.x$greaterSalePrice == 1, 1, 0))/nrow(prob.y.x)
b
```

```
## [1] 0.2342466
```

2.3 c. $P(X < x|Y > y)$

```
c <- (sum(ifelse(prob.y.x$lesserLotArea == 1 &
                   prob.y.x$greaterSalePrice == 1, 1, 0))
      / nrow(prob.y.x)) / ((sum(prob.y.x$lesserLotArea) / nrow(prob.y.x)))
c
```

```
## [1] 0.6876712
```

Does splitting the training data in this fashion make them independent?

In other words, does $P(X|Y) = P(X)P(Y)$?

> I am understanding this to mean does the probability of X>x given Y>y, which was answered for in part a. above, equal the probability of X>x mutiplied by Y>y

2.4 Mathematical Check for $P(X|Y) = P(X)P(Y)$

```
X <- sum(prob.y.x$greaterLotArea)/ nrow(prob.y.x)
Y <- sum(prob.y.x$greaterSalePrice) / nrow(prob.y.x)
X * Y
```

```
## [1] 0.1875
```

```
a == (X * Y)
```

```
## [1] FALSE
```

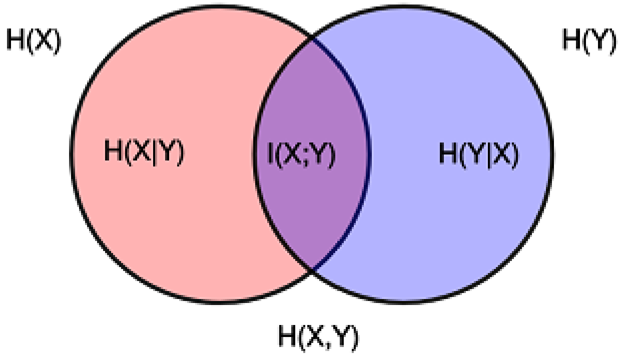
2.5 Chi Square test for association.

```r
prob.table <- as.data.frame(rbind(cbind(sum(prob.y.x$lesserLotArea),
                                        sum(prob.y.x$greaterLotArea)),
                                  cbind(sum(prob.y.x$lesserSalePrice),
                                        sum(prob.y.x$greaterSalePrice))))
chisq.test(prob.table)
```

```
##
##  Pearson's Chi-squared test with Yates' continuity correction
##
## data:  prob.table
## X-squared = 728, df = 1, p-value < 2.2e-16
```

> We see that the p-value is quite low, lower than the assumptive .05, so we therefore reject the null hypothesis that the values are independent of each other.

The below venn diagram from Wikipedia may provide a clearer understanding of the differences in these measures:



[1]

[1] By KonradVoelkel (Own work) [Public domain], via Wikimedia Commons

3 Descriptive and Inferential Statistics.

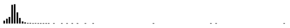
3.1 Univariate descriptive statistics and plots

3.1.1 Lot Area

```r
description <- describe(sub.train.df["LotArea"],descript = "LotArea")
latex(description, file = '')
```

LotArea
1 Variables 1460 Observations

LotArea



n	missing	distinct	Info	Mean	Gmd	.05	.10	.25	.50	.75	.90	.95
1460	0	1073	1	10517	5718	3312	5000	7554	9478	11602	14382	17401

```
lowest :   1300    1477    1491    1526    1533, highest:  70761 115149 159000 164660 215245
```

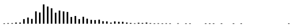
The histogram in the upper right corner of the table shows a right skewed distribution, which is not surprising since houses in cities would likely have similar lot areas versus instances of varied rural large lot areas.

3.1.2 Sale Price

```r
description <- describe(sub.train.df["SalePrice"], descript = "SalePrice")
latex(description, file = '')
```

SalePrice
1 Variables 1460 Observations

SalePrice



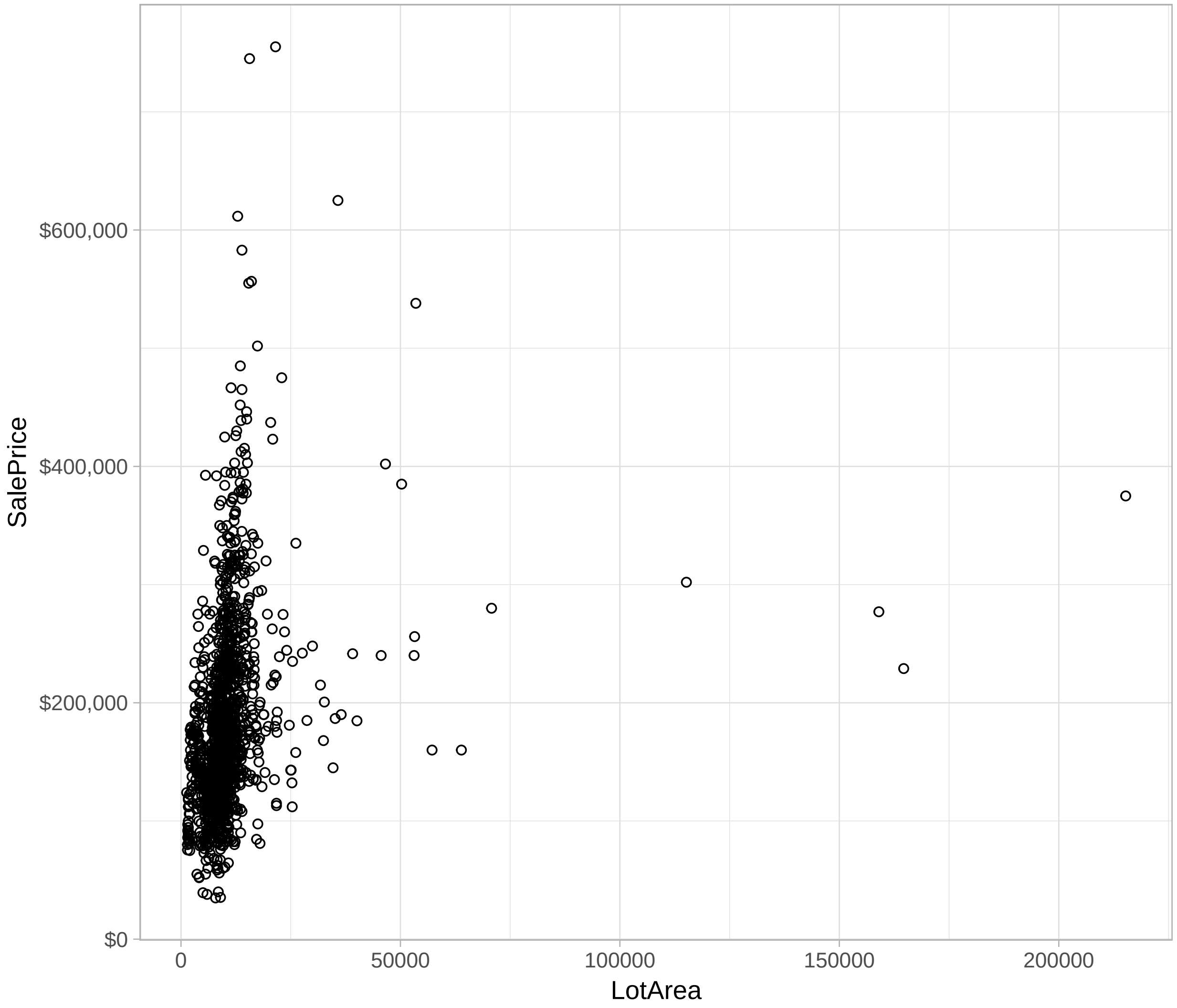
n	missing	distinct	Info	Mean	Gmd	.05	.10	.25	.50	.75	.90	.95
1460	0	663	1	180921	81086	88000	106475	129975	163000	214000	278000	326100

```
lowest :  34900  35311  37900  39300  40000, highest: 582933 611657 625000 745000 755000
```

As we can see from the histogram the shape of the data is near normal. It is interesting to visualize that lot area does not follow the same shape, this would hold with our original assumption that where the house is located has more impact than the size of the lot area.

3.2 Scatterplot of X and Y

```
ggplot(sub.train.df, aes(x = LotArea, y = SalePrice)) +
    geom_point(shape=1) +
    theme_light() +
    scale_y_continuous(labels = dollar)
```



I expected this type of scatter plot based on my anecdotal assumption of cities vs rural areas. I would think that the differences within the city limits have high variance in price but less variation in lot area. My assumption is that the large lot areas at the mid sale price is likely in a rural area.

3.3 Box-Cox transformations.

I am using the `BoxCox.lambda` function from the `forecast` package to determine the necessary transformations for the two variables.

```r
library(forecast)
library(knitr)
l1 <- BoxCox.lambda(as.numeric(sub.train.df$SalePrice))
l2 <- BoxCox.lambda(as.numeric(sub.train.df$LotArea))

lamdas <- c(l1, l2)
Variables <- c("SalePrice", "LotArea")
dfBoxCox <- as.data.frame(cbind(round(as.numeric(lamdas),4), Variables))
colnames(dfBoxCox) <- c("$\\lambda$", "Variables")
kable(dfBoxCox, align = c("c", "c"))
```

λ	Variables
-0.3308	SalePrice
-0.1268	LotArea

Common Box-Cox Transformations[2] [3]

λ	Y'
-0.5	$Y^{-0.5} = \frac{1}{\sqrt{(Y)}}$
0	$\log(Y)$
.25	$\sqrt[4]{Y}$

***

Lambda values were truncated to the nearest tenth
that match a common transformation as per the below table.

variable	variable transformation
SalePrice	$SalePrice^{-0.5}$
LotArea	$log(LotArea)$

[2] Osborne, Jason W. "Improving your data transformations: Applying the Box-Cox transformation." Practical Assessment, Research & Evaluation 15.12 (2010): 1-9.

[3] By Understanding Both the Concept of Transformation and the Box-Cox Method, Practitioners Will Be Better Prepared to Work with Non-normal Data. . "Making Data Normal Using Box-Cox Power Transformation." ISixSigma. N.p., n.d. Web. 29 Oct. 2016.

3.4 Correlation Analysis

3.4.1 Correlation analysis and interpetation

```
sub.train.df.trans <- sub.train.df %>%
                      mutate(SalePrice = SalePrice^(-.5),
                             LotArea = log(LotArea))

sub.train.cor <- cor.test(sub.train.df.trans$SalePrice,
                          sub.train.df.trans$LotArea,
                          method = "pearson", conf.level = .99)
sub.train.cor
```

```
##
##  Pearson's product-moment correlation
##
## data:  sub.train.df.trans$SalePrice and sub.train.df.trans$LotArea
## t = -15.968, df = 1458, p-value < 2.2e-16
## alternative hypothesis: true correlation is not equal to 0
## 99 percent confidence interval:
##  -0.4417063 -0.3269282
## sample estimates:
##        cor
## -0.3858091
```

The p-value of the correlation test is 2.2e-16 which is less than the significance level of alpha at .05. We are using the standard alpha as there is no indication another any other value for alpha should be used. We can therefore say that the log of lot size and sale price raised to the -.5 power are significantly correlated with a negative correlation coefficient of -0.386.

3.4.2 Null hypothesis test at a 99% confidence interval.

The correlation test has specifically done that for us and we can safely reject the null hypothesis as we see that our 99% confidence interval exists at the values (-0.441, -0.327) with a p-value < 2.2e-16.

3.4.3 Analysis Discussion

This means two possible things could have occured, there is no correlation and this data set is pulled from an unusual set of house sales. Or, more likely with the values obtained, our assumption of 0 correlation is incorect and we have obtained a very typical data set and must reject the null hypothesis because correlation does exist.

4 Linear Algebra and Correlation.

4.1 Correlation Matrix

```
A <- cor(sub.train.df.trans)
kable(A)
```

	SalePrice	LotArea
SalePrice	1.0000000	-0.3858091
LotArea	-0.3858091	1.0000000

4.2 Inverted correlation matrix (percision matrix)

```
B <- solve(A)
kable(B)
```

	SalePrice	LotArea
SalePrice	1.1748792	0.4532792
LotArea	0.4532792	1.1748792

4.3 Multiply the correlation matrix by the precision matrix, and then multiply the precision matrix by the correlation matrix.

```
corr.by.pre.M <- A %*% B
kable(corr.by.pre.M)
```

	SalePrice	LotArea
SalePrice	1	0
LotArea	0	1

```
pre.by.corr.M <- B %*% A
kable(pre.by.corr.M)
```

	SalePrice	LotArea
SalePrice	1	0
LotArea	0	1

5 Calculus-Based Probability & Statistics

5.1 Location Shift

Many times, it makes sense to fit a closed form distribution to data. For your non-transformed independent variable, location shift it so that the minimum value is above zero.

```
min(sub.train.df$LotArea)
```

```
[1] 1300
```

> For the independent variable chosen, there are no zero values observed. This makes sense as we would expect the lot area to have some value and I would expect it to never be unobserved (an assumption that at least estimates would be used without a true figure).
>
> However, if a shift was required something like the below could be used.

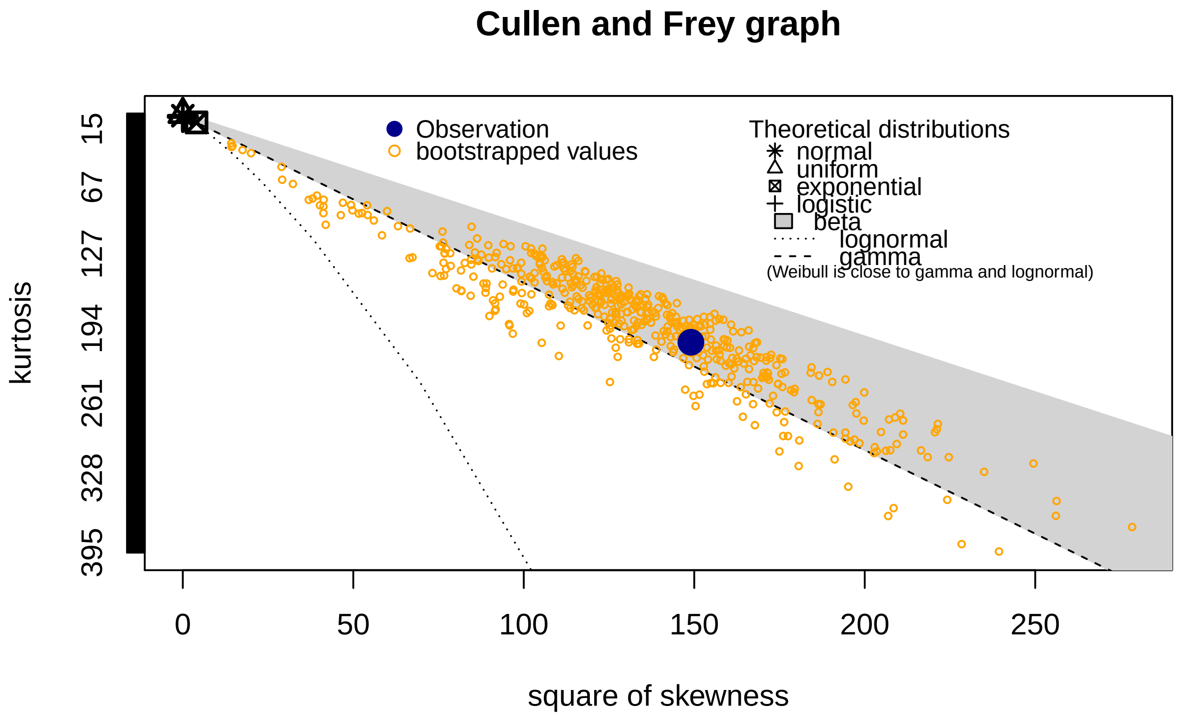
```
shift <- sub.train.df$LotArea + 1
```

5.2 Mass and `fitdistr`

Then load the MASS package and run fitdistr to fit a density function of your choice. (See https://stat.ethz.ch/R-manual/R-devel/library/MASS/html/fitdistr.html).

5.2.1 Best fit distribution.

```
library(fitdistrplus)
descdist(sub.train.df$LotArea, discrete=FALSE, boot=500)
```



```
## summary statistics
## ------
## min:  1300   max:  215245
## median:  9478.5
## mean:  10516.83
## estimated sd:  9981.265
## estimated skewness:  12.20769
## estimated kurtosis:  206.2433
```

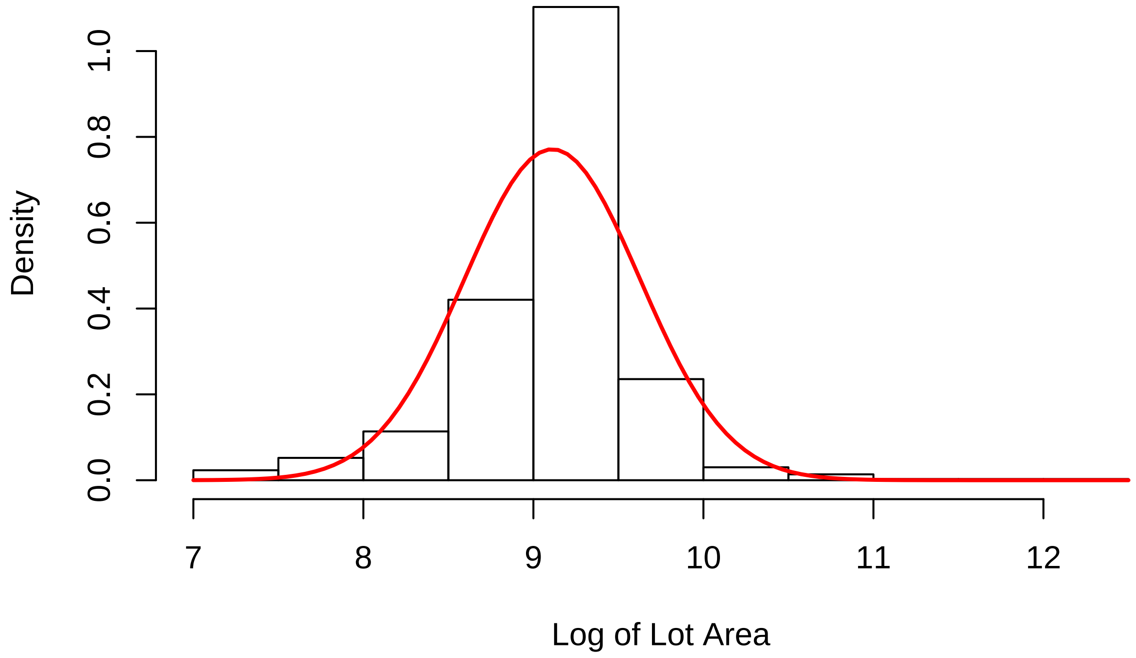
5.3 Fitting Distribution

There were too many issues in attempting to fit the `beta` distribution so the next best theoretical distribution was used - `log normal`.

```
library(MASS)
fit.log <- fitdistr(sub.train.df$LotArea, densfun = "log-normal")
fit.log
```

```
##      meanlog         sdlog
##   9.110838240   0.517270830
##  (0.013537596) (0.009572526)
```

```
hist(log(sub.train.df$LotArea), prob=TRUE, xlab = "Log of Lot Area", main = "")
curve(dnorm(x, fit.log$estimate[1], fit.log$estimate[2]), col="red", lwd=2, add=T)
```



Our density plot indicates that the log normal distribution fits quite well.

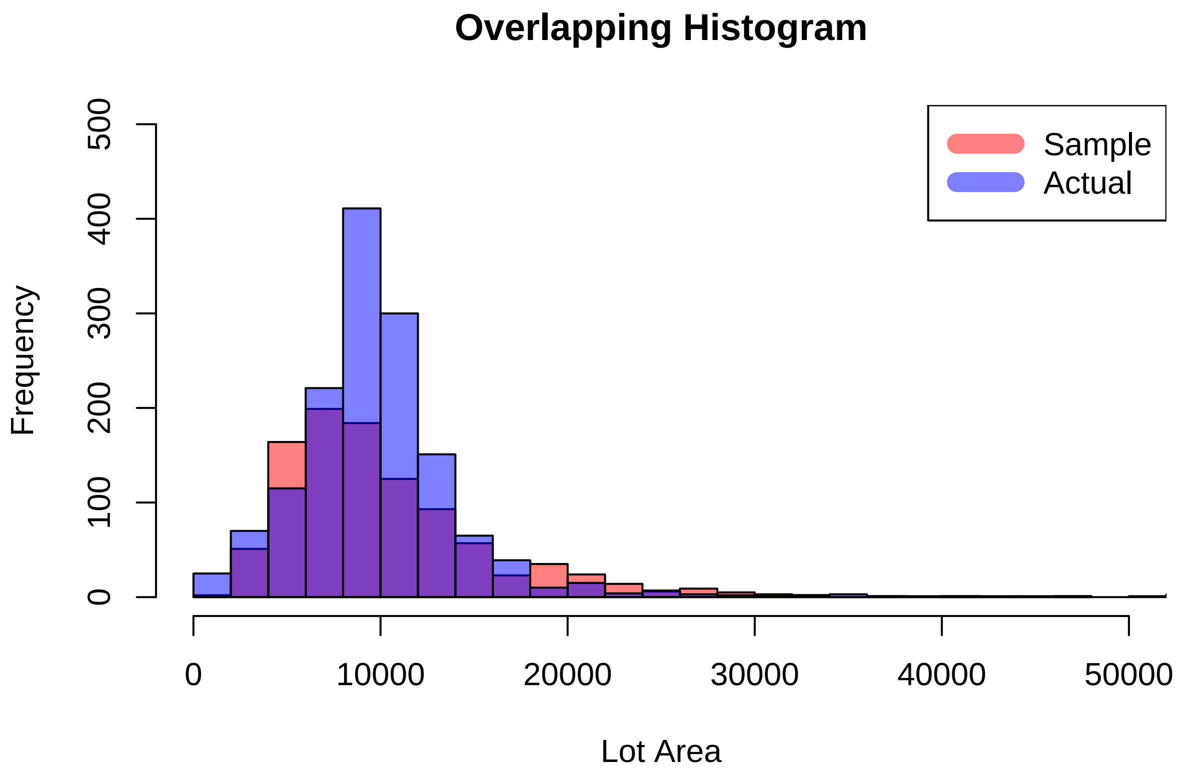
5.4 Sampling

Find the optimal value of the parameters for this distribution, and then take 1000 samples from this distribution (e.g., rexp(1000) for an exponential).

```
set.seed(1234)
sample <- rlnorm(1000, meanlog = fit.log$estimate[1], sdlog = fit.log$estimate[2])
```

Plot a histogram and compare it with a histogram of your non-transformed original variable.

```
hist(sample, pch = 20, breaks = 25, col = rgb(1,0,0,0.5),
     xlim = c(0,50000), ylim = c(0,500),
     main = 'Overlapping Histogram', xlab = 'Lot Area')
hist(sub.train.df$LotArea, pch = 20, breaks = 100,
     col = rgb(0,0,1,0.5), add = T)
#https://www.r-bloggers.com/overlapping-histogram-in-r/
legend("topright", c("Sample", "Actual"), col=c(rgb(1,0,0,0.5), rgb(0,0,1,0.5)), lwd=10)
```



It is clear that the distributions are very similar. Plotting them together gives a clear visual of how similar the distributions are and note that x has been limited to not extend to extreme values.

6 Modeling

6.1 Regression model.

6.1.1 Variable Exploration

```
description <- describe(train.df %>% dplyr::select(-Id, -SalePrice), descript = "Training Data Set")
latex(description, file = '')
```

Training Data Set
79 Variables 1460 Observations

MSSubClass

```
    n  missing  distinct   Info   Mean    Gmd   .05   .10   .25   .50   .75   .90   .95
 1460        0        15   0.94   56.9  43.19    20    20    20    50    70   120   160

Value         20    30    40    45    50    60    70    75    80    85    90   120   160   180
Frequency    536    69     4    12   144   299    60    16    58    20    52    87    63    10
Proportion 0.367 0.047 0.003 0.008 0.099 0.205 0.041 0.011 0.040 0.014 0.036 0.060 0.043 0.007

Value        190
Frequency     30
Proportion 0.021
```

MSZoning

```
    n  missing  distinct
 1460        0         5

Value      C (all)      FV      RH      RL      RM
Frequency       10      65      16    1151     218
Proportion   0.007   0.045   0.011   0.788   0.149
```

LotFrontage

```
    n  missing  distinct   Info   Mean    Gmd   .05   .10   .25   .50   .75   .90   .95
 1201      259       110  0.998  70.05  24.61    34    44    59    69    80    96   107

lowest :  21  24  30  32  33, highest: 160 168 174 182 313
```

LotArea

```
    n  missing  distinct   Info   Mean    Gmd    .05    .10    .25    .50    .75    .90    .95
 1460        0      1073      1  10517   5718   3312   5000   7554   9478  11602  14382  17401

lowest :    1300    1477    1491    1526    1533, highest:  70761 115149 159000 164660 215245
```

Street

```
    n  missing  distinct
 1460        0         2

Value       Grvl  Pave
Frequency      6  1454
Proportion 0.004 0.996
```

Alley

```
  n  missing  distinct
 91     1369         2

Value       Grvl  Pave
Frequency     50    41
Proportion 0.549 0.451
```

LotShape

```
    n  missing  distinct
 1460        0         4

Value        IR1   IR2   IR3   Reg
Frequency    484    41    10   925
Proportion 0.332 0.028 0.007 0.634
```

LandContour

```
     n  missing  distinct
  1460        0         4

Value        Bnk   HLS   Low   Lvl
Frequency     63    50    36  1311
Proportion 0.043 0.034 0.025 0.898
```

Utilities

```
     n  missing  distinct
  1460        0         2

Value      AllPub NoSeWa
Frequency    1459      1
Proportion  0.999  0.001
```

LotConfig

```
     n  missing  distinct
  1460        0         5

Value       Corner CulDSac     FR2     FR3  Inside
Frequency      263      94      47       4    1052
Proportion   0.180   0.064   0.032   0.003   0.721
```

LandSlope

```
     n  missing  distinct
  1460        0         3

Value        Gtl   Mod   Sev
Frequency   1382    65    13
Proportion 0.947 0.045 0.009
```

Neighborhood

```
     n  missing  distinct
  1460        0        25

lowest : Blmngtn Blueste BrDale  BrkSide ClearCr, highest: Somerst StoneBr SWISU   Timber  Veenker
```

Condition1

```
     n  missing  distinct
  1460        0         9

Value      Artery  Feedr   Norm   PosA   PosN   RRAe   RRAn   RRNe   RRNn
Frequency      48     81   1260      8     19     11     26      2      5
Proportion  0.033  0.055  0.863  0.005  0.013  0.008  0.018  0.001  0.003
```

Condition2

```
     n  missing  distinct
  1460        0         8

Value      Artery  Feedr   Norm   PosA   PosN   RRAe   RRAn   RRNn
Frequency       2      6   1445      1      2      1      1      2
Proportion  0.001  0.004  0.990  0.001  0.001  0.001  0.001  0.001
```

BldgType

```
     n  missing  distinct
  1460        0         5

Value        1Fam 2fmCon Duplex  Twnhs TwnhsE
Frequency    1220     31     52     43    114
Proportion  0.836  0.021  0.036  0.029  0.078
```

HouseStyle

```
     n  missing  distinct
  1460        0         8

Value      1.5Fin 1.5Unf 1Story 2.5Fin 2.5Unf 2Story SFoyer   SLvl
Frequency     154     14    726      8     11    445     37     65
Proportion  0.105  0.010  0.497  0.005  0.008  0.305  0.025  0.045
```

OverallQual

```
     n  missing  distinct     Info     Mean      Gmd      .05      .10      .25      .50      .75      .90      .95
  1460        0        10    0.951    6.099    1.522        4        5        5        6        7        8        8

Value          1     2     3     4     5     6     7     8     9    10
Frequency      2     3    20   116   397   374   319   168    43    18
Proportion 0.001 0.002 0.014 0.079 0.272 0.256 0.218 0.115 0.029 0.012
```

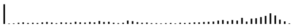
OverallCond

```
      n  missing  distinct     Info    Mean    Gmd
   1460        0         9    0.814   5.575  1.111

Value          1     2     3     4     5     6     7     8     9
Frequency      1     5    25    57   821   252   205    72    22
Proportion 0.001 0.003 0.017 0.039 0.562 0.173 0.140 0.049 0.015
```

YearBuilt

```
      n  missing  distinct    Info    Mean     Gmd     .05     .10     .25     .50     .75     .90     .95
   1460        0       112       1    1971   33.88    1916    1925    1954    1973    2000    2006    2007

lowest : 1872 1875 1880 1882 1885, highest: 2006 2007 2008 2009 2010
```

YearRemodAdd



```
      n  missing  distinct     Info    Mean     Gmd     .05     .10     .25     .50     .75     .90     .95
   1460        0        61    0.997    1985   23.05    1950    1950    1967    1994    2004    2006    2007

lowest : 1950 1951 1952 1953 1954, highest: 2006 2007 2008 2009 2010
```

RoofStyle

```
      n  missing  distinct
   1460        0         6

Value         Flat   Gable Gambrel     Hip Mansard    Shed
Frequency       13    1141      11     286       7       2
Proportion   0.009   0.782   0.008   0.196   0.005   0.001
```

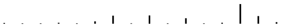
RoofMatl

```
      n  missing  distinct
   1460        0         8

Value      ClyTile CompShg Membran   Metal    Roll Tar&Grv WdShake WdShngl
Frequency        1    1434       1       1       1      11       5       6
Proportion   0.001   0.982   0.001   0.001   0.001   0.008   0.003   0.004
```

Exterior1st

```
      n  missing  distinct
   1460        0        15

Value      AsbShng AsphShn BrkComm BrkFace  CBlock CemntBd HdBoard ImStucc MetalSd Plywood
Frequency       20       1       2      50       1      61     222       1     220     108
Proportion   0.014   0.001   0.001   0.034   0.001   0.042   0.152   0.001   0.151   0.074

Value        Stone  Stucco VinylSd Wd Sdng WdShing
Frequency        2      25     515     206      26
Proportion   0.001   0.017   0.353   0.141   0.018
```

Exterior2nd



```
      n  missing  distinct
   1460        0        16

Value      AsbShng AsphShn Brk Cmn BrkFace  CBlock CmentBd HdBoard ImStucc MetalSd   Other
Frequency       20       3       7      25       1      60     207      10     214       1
Proportion   0.014   0.002   0.005   0.017   0.001   0.041   0.142   0.007   0.147   0.001

Value      Plywood   Stone  Stucco VinylSd Wd Sdng Wd Shng
Frequency      142       5      26     504     197      38
Proportion   0.097   0.003   0.018   0.345   0.135   0.026
```

MasVnrType

```
      n  missing  distinct
   1452        8         4

Value       BrkCmn BrkFace    None   Stone
Frequency       15     445     864     128
Proportion   0.010   0.306   0.595   0.088
```

MasVnrArea

```
      n  missing  distinct     Info    Mean    Gmd    .05    .10    .25    .50    .75    .90    .95
   1452        8       327    0.791   103.7  156.9      0      0      0      0    166    335    456

lowest :    0    1   11   14   16, highest: 1115 1129 1170 1378 1600
```

ExterQual

```
      n  missing  distinct
   1460        0         4

Value         Ex    Fa    Gd    TA
Frequency     52    14   488   906
Proportion 0.036 0.010 0.334 0.621
```

ExterCond

```
     n  missing  distinct 
  1460        0         5 

Value         Ex    Fa    Gd    Po    TA
Frequency      3    28   146     1  1282
Proportion 0.002 0.019 0.100 0.001 0.878
```

Foundation

```
     n  missing  distinct 
  1460        0         6 

Value      BrkTil CBlock  PConc   Slab  Stone   Wood
Frequency     146    634    647     24      6      3
Proportion  0.100  0.434  0.443  0.016  0.004  0.002
```

BsmtQual

```
     n  missing  distinct 
  1423       37         4 

Value         Ex    Fa    Gd    TA
Frequency    121    35   618   649
Proportion 0.085 0.025 0.434 0.456
```

BsmtCond

```
     n  missing  distinct 
  1423       37         4 

Value         Fa    Gd    Po    TA
Frequency     45    65     2  1311
Proportion 0.032 0.046 0.001 0.921
```

BsmtExposure

```
     n  missing  distinct 
  1422       38         4 

Value         Av    Gd    Mn    No
Frequency    221   134   114   953
Proportion 0.155 0.094 0.080 0.670
```

BsmtFinType1

```
     n  missing  distinct 
  1423       37         6 

Value        ALQ   BLQ   GLQ   LwQ   Rec   Unf
Frequency    220   148   418    74   133   430
Proportion 0.155 0.104 0.294 0.052 0.093 0.302
```

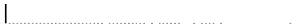
BsmtFinSF1

```
     n  missing  distinct     Info     Mean      Gmd      .05      .10      .25      .50      .75      .90      .95 
  1460        0       637    0.967    443.6    484.5      0.0      0.0      0.0    383.5    712.2   1065.5   1274.0 

lowest :    0    2   16   20   24, highest: 1904 2096 2188 2260 5644
```

BsmtFinType2

```
     n  missing  distinct 
  1422       38         6 

Value        ALQ   BLQ   GLQ   LwQ   Rec   Unf
Frequency     19    33    14    46    54  1256
Proportion 0.013 0.023 0.010 0.032 0.038 0.883
```

BsmtFinSF2



```
     n  missing  distinct     Info     Mean      Gmd      .05      .10      .25      .50      .75      .90      .95 
  1460        0       144    0.305    46.55    86.58      0.0      0.0      0.0      0.0      0.0    117.2    396.2 

lowest :    0   28   32   35   40, highest: 1080 1085 1120 1127 1474
```

BsmtUnfSF

```
     n  missing  distinct     Info     Mean      Gmd      .05      .10      .25      .50      .75      .90      .95 
  1460        0       780    0.999    567.2    486.6      0.0     74.9    223.0    477.5    808.0   1232.0   1468.0 

lowest :    0   14   15   23   26, highest: 2042 2046 2121 2153 2336
```

TotalBsmtSF

```
     n  missing  distinct     Info     Mean      Gmd      .05      .10      .25      .50      .75      .90      .95 
  1460        0       721        1     1057    459.5    519.3    636.9    795.8    991.5   1298.2   1602.2   1753.0 

lowest :    0  105  190  264  270, highest: 3094 3138 3200 3206 6110
```

Heating

```
      n  missing  distinct
   1460        0         6

Value      Floor  GasA  GasW  Grav  OthW  Wall
Frequency      1  1428    18     7     2     4
Proportion 0.001 0.978 0.012 0.005 0.001 0.003
```

HeatingQC

```
      n  missing  distinct
   1460        0         5

Value         Ex    Fa    Gd    Po    TA
Frequency    741    49   241     1   428
Proportion 0.508 0.034 0.165 0.001 0.293
```

CentralAir

```
      n  missing  distinct
   1460        0         2

Value          N     Y
Frequency     95  1365
Proportion 0.065 0.935
```

Electrical

```
      n  missing  distinct
   1459        1         5

Value      FuseA FuseF FuseP   Mix SBrkr
Frequency     94    27     3     1  1334
Proportion 0.064 0.019 0.002 0.001 0.914
```

X1stFlrSF

```
      n   missing   distinct   Info   Mean    Gmd     .05     .10     .25     .50     .75     .90     .95
   1460         0        753      1   1163  416.4   673.0   756.9   882.0  1087.0  1391.2  1680.0  1831.2

lowest :  334  372  438  480  483, highest: 2633 2898 3138 3228 4692
```

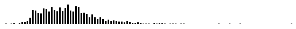
X2ndFlrSF

```
      n   missing   distinct   Info   Mean    Gmd   .05   .10   .25   .50    .75    .90     .95
   1460         0        417  0.817    347  450.2   0.0   0.0   0.0   0.0  728.0  954.2  1141.0

lowest :    0  110  167  192  208, highest: 1611 1796 1818 1872 2065
```

LowQualFinSF

```
      n  missing  distinct   Info   Mean    Gmd  .05  .10  .25  .50  .75  .90  .95
   1460        0        24  0.052  5.845  11.55    0    0    0    0    0    0    0

lowest :   0  53  80 120 144, highest: 513 514 515 528 572
```

GrLivArea



```
      n  missing  distinct  Info  Mean    Gmd  .05  .10   .25   .50   .75   .90   .95
   1460        0       861     1  1515  563.1  848  912  1130  1464  1777  2158  2466

lowest :  334  438  480  520  605, highest: 3627 4316 4476 4676 5642
```

BsmtFullBath

```
      n  missing  distinct   Info    Mean     Gmd
   1460        0         4  0.733  0.4253  0.5085

Value          0     1     2     3
Frequency    856   588    15     1
Proportion 0.586 0.403 0.010 0.001
```

BsmtHalfBath

```
      n  missing  distinct   Info     Mean     Gmd
   1460        0         3  0.159  0.05753  0.1088

Value          0     1     2
Frequency   1378    80     2
Proportion 0.944 0.055 0.001
```

FullBath

```
      n  missing  distinct   Info   Mean     Gmd
   1460        0         4  0.766  1.565  0.5521

Value          0     1     2     3
Frequency      9   650   768    33
Proportion 0.006 0.445 0.526 0.023
```

HalfBath

```
      n  missing distinct     Info     Mean      Gmd 
   1460        0        3    0.706   0.3829   0.4852 

Value          0     1     2
Frequency    913   535    12
Proportion 0.625 0.366 0.008
```

BedroomAbvGr

```
      n  missing distinct     Info     Mean      Gmd 
   1460        0        8    0.815    2.866    0.818 

Value          0     1     2     3     4     5     6     8
Frequency      6    50   358   804   213    21     7     1
Proportion 0.004 0.034 0.245 0.551 0.146 0.014 0.005 0.001
```

KitchenAbvGr

```
      n  missing distinct     Info     Mean      Gmd 
   1460        0        4    0.133    1.047  0.09174 

Value          0     1     2     3
Frequency      1  1392    65     2
Proportion 0.001 0.953 0.045 0.001
```

KitchenQual

```
      n  missing distinct 
   1460        0        4 

Value         Ex    Fa    Gd    TA
Frequency    100    39   586   735
Proportion 0.068 0.027 0.401 0.503
```

TotRmsAbvGrd

```
      n  missing distinct     Info     Mean      Gmd      .05      .10      .25      .50      .75      .90      .95 
   1460        0       12    0.958    6.518    1.762        4        5        5        6        7        9       10 

Value          2     3     4     5     6     7     8     9    10    11    12    14
Frequency      1    17    97   275   402   329   187    75    47    18    11     1
Proportion 0.001 0.012 0.066 0.188 0.275 0.225 0.128 0.051 0.032 0.012 0.008 0.001
```

Functional

```
      n  missing distinct 
   1460        0        7 

Value       Maj1  Maj2  Min1  Min2   Mod   Sev   Typ
Frequency     14     5    31    34    15     1  1360
Proportion 0.010 0.003 0.021 0.023 0.010 0.001 0.932
```

Fireplaces

```
      n  missing distinct     Info     Mean      Gmd 
   1460        0        4    0.806    0.613   0.6566 

Value          0     1     2     3
Frequency    690   650   115     5
Proportion 0.473 0.445 0.079 0.003
```

FireplaceQu

```
     n  missing distinct 
   770      690        5 

Value         Ex    Fa    Gd    Po    TA
Frequency     24    33   380    20   313
Proportion 0.031 0.043 0.494 0.026 0.406
```

GarageType

```
      n  missing distinct 
   1379       81        6 

Value       2Types  Attchd Basment BuiltIn CarPort  Detchd
Frequency        6     870      19      88       9     387
Proportion   0.004   0.631   0.014   0.064   0.007   0.281
```

GarageYrBlt

```
      n  missing distinct     Info     Mean      Gmd      .05      .10      .25      .50      .75      .90      .95 
   1379       81       97        1     1979    27.63     1930     1945     1961     1980     2002     2006     2007 

lowest : 1900 1906 1908 1910 1914, highest: 2006 2007 2008 2009 2010
```

GarageFinish

```
       n  missing  distinct
    1379       81         3

Value        Fin   RFn   Unf
Frequency    352   422   605
Proportion 0.255 0.306 0.439
```

GarageCars

```
       n  missing  distinct     Info     Mean      Gmd
    1460        0         5    0.802    1.767   0.7609

Value          0     1     2     3     4
Frequency     81   369   824   181     5
Proportion 0.055 0.253 0.564 0.124 0.003
```

GarageArea

```
       n  missing  distinct     Info     Mean      Gmd      .05      .10      .25      .50      .75      .90      .95
    1460        0       441        1      473    234.9      0.0    240.0    334.5    480.0    576.0    757.1    850.1

lowest :    0  160  164  180  186, highest: 1220 1248 1356 1390 1418
```

GarageQual

```
       n  missing  distinct
    1379       81         5

Value         Ex    Fa    Gd    Po    TA
Frequency      3    48    14     3  1311
Proportion 0.002 0.035 0.010 0.002 0.951
```

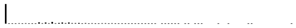
GarageCond

```
       n  missing  distinct
    1379       81         5

Value         Ex    Fa    Gd    Po    TA
Frequency      2    35     9     7  1326
Proportion 0.001 0.025 0.007 0.005 0.962
```

PavedDrive

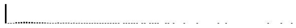
```
       n  missing  distinct
    1460        0         3

Value          N     P     Y
Frequency     90    30  1340
Proportion 0.062 0.021 0.918
```

WoodDeckSF



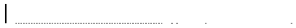
```
       n  missing  distinct     Info     Mean      Gmd      .05      .10      .25      .50      .75      .90      .95
    1460        0       274    0.858    94.24      125        0        0        0        0      168      262      335

lowest :   0  12  24  26  28, highest: 668 670 728 736 857
```

OpenPorchSF



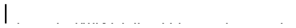
```
       n  missing  distinct     Info     Mean      Gmd      .05      .10      .25      .50      .75      .90      .95
    1460        0       202    0.909    46.66    62.43        0        0        0       25       68      130      175

lowest :   0   4   8  10  11, highest: 406 418 502 523 547
```

EnclosedPorch



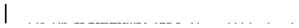
```
       n  missing  distinct     Info     Mean      Gmd      .05      .10      .25      .50      .75      .90      .95
    1460        0       120    0.369    21.95    39.39      0.0      0.0      0.0      0.0      0.0    112.0    180.1

lowest :   0  19  20  24  30, highest: 301 318 330 386 552
```

X3SsnPorch



```
       n  missing  distinct     Info     Mean      Gmd      .05      .10      .25      .50      .75      .90      .95
    1460        0        20    0.049     3.41    6.739        0        0        0        0        0        0        0

Value          0    23    96   130   140   144   153   162   168   180   182   196   216   238
Frequency   1436     1     1     1     1     2     1     1     3     2     1     1     2     1
Proportion 0.984 0.001 0.001 0.001 0.001 0.001 0.001 0.001 0.002 0.001 0.001 0.001 0.001 0.001

Value        245   290   304   320   407   508
Frequency      1     1     1     1     1     1
Proportion 0.001 0.001 0.001 0.001 0.001 0.001
```

ScreenPorch



```
       n  missing  distinct     Info     Mean      Gmd      .05      .10      .25      .50      .75      .90      .95
    1460        0        76     0.22    15.06    28.27        0        0        0        0        0        0      160

lowest :   0  40  53  60  63, highest: 385 396 410 440 480
```

PoolArea

```
       n  missing  distinct     Info     Mean      Gmd
    1460        0         8    0.014    2.759    5.497

Value          0   480   512   519   555   576   648   738
Frequency   1453     1     1     1     1     1     1     1
Proportion 0.995 0.001 0.001 0.001 0.001 0.001 0.001 0.001
```

PoolQC

```
   n  missing  distinct
   7     1453         3

Value         Ex    Fa    Gd
Frequency      2     2     3
Proportion 0.286 0.286 0.429
```

Fence

```
     n  missing  distinct
   281     1179         4

Value      GdPrv  GdWo MnPrv  MnWw
Frequency     59    54   157    11
Proportion 0.210 0.192 0.559 0.039
```

MiscFeature

```
    n  missing  distinct
   54     1406         4

Value       Gar2  Othr  Shed  TenC
Frequency      2     2    49     1
Proportion 0.037 0.037 0.907 0.019
```

MiscVal

```
       n  missing  distinct     Info     Mean      Gmd      .05      .10      .25      .50      .75      .90      .95
    1460        0        21    0.103    43.49    85.67        0        0        0        0        0        0        0

Value          0    50   350   400   450   500   550   600   700   800  1150  1200  1300  1400
Frequency   1408     1     1    11     4    10     1     5     5     1     1     2     1     1
Proportion 0.964 0.001 0.001 0.008 0.003 0.007 0.001 0.003 0.003 0.001 0.001 0.001 0.001 0.001

Value       2000  2500  3500  8300 15500
Frequency      4     1     1     1     1
Proportion 0.003 0.001 0.001 0.001 0.001
```

MoSold

```
       n  missing  distinct     Info     Mean      Gmd      .05      .10      .25      .50      .75      .90      .95
    1460        0        12    0.985    6.322    3.041        2        3        5        6        8       10       11

Value          1     2     3     4     5     6     7     8     9    10    11    12
Frequency     58    52   106   141   204   253   234   122    63    89    79    59
Proportion 0.040 0.036 0.073 0.097 0.140 0.173 0.160 0.084 0.043 0.061 0.054 0.040
```

YrSold

```
       n  missing  distinct     Info     Mean      Gmd
    1460        0         5    0.955     2008    1.498

Value       2006  2007  2008  2009  2010
Frequency    314   329   304   338   175
Proportion 0.215 0.225 0.208 0.232 0.120
```

SaleType

```
       n  missing  distinct
    1460        0         9

Value        COD   Con ConLD ConLI ConLw   CWD   New   Oth    WD
Frequency     43     2     9     5     5     4   122     3  1267
Proportion 0.029 0.001 0.006 0.003 0.003 0.003 0.084 0.002 0.868
```

SaleCondition

```
       n  missing  distinct
    1460        0         6

Value      Abnorml AdjLand  Alloca  Family  Normal Partial
Frequency      101       4      12      20    1198     125
Proportion   0.069   0.003   0.008   0.014   0.821   0.086
```

6.2 Variable Selection

In the previous tables we see a number of variables with NAs. For simplicity, we will exclude the columns containing NAs and attempt to identify the variables of greatest importance.

6.3 Variable Subset Matrix

```
library(leaps)
train.df.2 <- train.df[ , colSums(is.na(train.df)) == 0]
subsetModel <- suppressMessages(regsubsets(SalePrice ~ ., data = train.df.2,
                        nbest = 1, really.big = T,  method = "seqrep"))
```

```
## Reordering variables and trying again:
```

```
summary <- summary(subsetModel, matrix = TRUE)
kable(t(summary$outmat))
```

	1(1)	2(1)	3(1)	4(1)	5(1)	6(1)	7(1)	8(1)	9(1)
Id									
MSSubClass						*	*	*	*
MSZoningFV									
MSZoningRH									
MSZoningRL									
MSZoningRM									
LotArea									
StreetPave									
LotShapeIR2									
LotShapeIR3									
LotShapeReg									
LandContourHLS									
LandContourLow									
LandContourLvl									
UtilitiesNoSeWa									
LotConfigCulDSac									
LotConfigFR2									
LotConfigFR3									
LotConfigInside									
LandSlopeMod									
LandSlopeSev									
NeighborhoodBlueste									
NeighborhoodBrDale									
NeighborhoodBrkSide									
NeighborhoodClearCr									
NeighborhoodCollgCr									
NeighborhoodCrawfor									
NeighborhoodEdwards									
NeighborhoodGilbert									
NeighborhoodIDOTRR									
NeighborhoodMeadowV									
NeighborhoodMitchel									
NeighborhoodNAmes									
NeighborhoodNoRidge								*	*
NeighborhoodNPkVill									
NeighborhoodNridgHt					*	*	*	*	*
NeighborhoodNWAmes									
NeighborhoodOldTown									
NeighborhoodSawyer									
NeighborhoodSawyerW									
NeighborhoodSomerst									
NeighborhoodStoneBr							*	*	*
NeighborhoodSWISU									
NeighborhoodTimber									
NeighborhoodVeenker									
Condition1Feedr									

	1 (1)	2 (1)	3 (1)	4 (1)	5 (1)	6 (1)	7 (1)	8 (1)	9 (1)
Condition1Norm									
Condition1PosA									
Condition1PosN									
Condition1RRAe									
Condition1RRAn									
Condition1RRNe									
Condition1RRNn									
Condition2Feedr									
Condition2Norm									
Condition2PosA									
Condition2PosN									
Condition2RRAe									
Condition2RRAn									
Condition2RRNn									
BldgType2fmCon									
BldgTypeDuplex									
BldgTypeTwnhs									
BldgTypeTwnhsE									
HouseStyle1.5Unf									
HouseStyle1Story									
HouseStyle2.5Fin									
HouseStyle2.5Unf									
HouseStyle2Story									
HouseStyleSFoyer									
HouseStyleSLvl									
OverallQual	*	*	*	*	*	*	*	*	*
OverallCond									
YearBuilt						*	*	*	
YearRemodAdd									*
RoofStyleGable									
RoofStyleGambrel									
RoofStyleHip									
RoofStyleMansard									
RoofStyleShed									
RoofMatlCompShg									
RoofMatlMembran									
RoofMatlMetal									
RoofMatlRoll									
RoofMatlTar&Grv									
RoofMatlWdShake									
RoofMatlWdShngl									
Exterior1stAsphShn									
Exterior1stBrkComm									
Exterior1stBrkFace									
Exterior1stCBlock									
Exterior1stCemntBd									
Exterior1stHdBoard									
Exterior1stImStucc									
Exterior1stMetalSd									
Exterior1stPlywood									
Exterior1stStone									
Exterior1stStucco									
Exterior1stVinylSd									
Exterior1stWd Sdng									
Exterior1stWdShing									
Exterior2ndAsphShn									
Exterior2ndBrk Cmn									
Exterior2ndBrkFace									
Exterior2ndCBlock									
Exterior2ndCmentBd									
Exterior2ndHdBoard									
Exterior2ndImStucc									
Exterior2ndMetalSd									
Exterior2ndOther									
Exterior2ndPlywood									
Exterior2ndStone									

	1 (1)	2 (1)	3 (1)	4 (1)	5 (1)	6 (1)	7 (1)	8 (1)	9 (1)
Exterior2ndStucco									
Exterior2ndVinylSd									
Exterior2ndWd Sdng									
Exterior2ndWd Shng									
ExterQualFa									
ExterQualGd									
ExterQualTA									
ExterCondFa									
ExterCondGd									
ExterCondPo									
ExterCondTA									
FoundationCBlock									
FoundationPConc									
FoundationSlab									
FoundationStone									
FoundationWood									
BsmtFinSF1			*	*	*	*	*	*	*
BsmtFinSF2									
BsmtUnfSF									
TotalBsmtSF									
HeatingGasA									
HeatingGasW									
HeatingGrav									
HeatingOthW									
HeatingWall									
HeatingQCFa									
HeatingQCGd									
HeatingQCPo									
HeatingQCTA									
CentralAirY									
X1stFlrSF									
X2ndFlrSF									
LowQualFinSF									
GrLivArea		*	*	*	*	*	*	*	*
BsmtFullBath									
BsmtHalfBath									
FullBath									
HalfBath									
BedroomAbvGr									
KitchenAbvGr									
KitchenQualFa									
KitchenQualGd									
KitchenQualTA									
TotRmsAbvGrd									
FunctionalMaj2									
FunctionalMin1									
FunctionalMin2									
FunctionalMod									
FunctionalSev									
FunctionalTyp									
Fireplaces									
GarageCars				*	*				*
GarageArea									
PavedDriveP									
PavedDriveY									
WoodDeckSF									
OpenPorchSF									
EnclosedPorch									
X3SsnPorch									
ScreenPorch									
PoolArea									
MiscVal									
MoSold									
YrSold									
SaleTypeCon									
SaleTypeConLD									

	1 (1)	2 (1)	3 (1)	4 (1)	5 (1)	6 (1)	7 (1)	8 (1)	9 (1)
SaleTypeConLI									
SaleTypeConLw									
SaleTypeCWD									
SaleTypeNew									
SaleTypeOth									
SaleTypeWD									
SaleConditionAdjLand									
SaleConditionAlloca									
SaleConditionFamily									
SaleConditionNormal									
SaleConditionPartial									

6.4 First Model

Based on the last best model we will limit the data set to the following variables `MSSubClass`, `NeighborhoodNoRidge`, `NeighborhoodNridgHt`, `NeighborhoodStoneBr`, `OverallQual`, `YearRemodAdd`, `BsmtFinSF1`, `GrLivArea`, `GarageCars`.

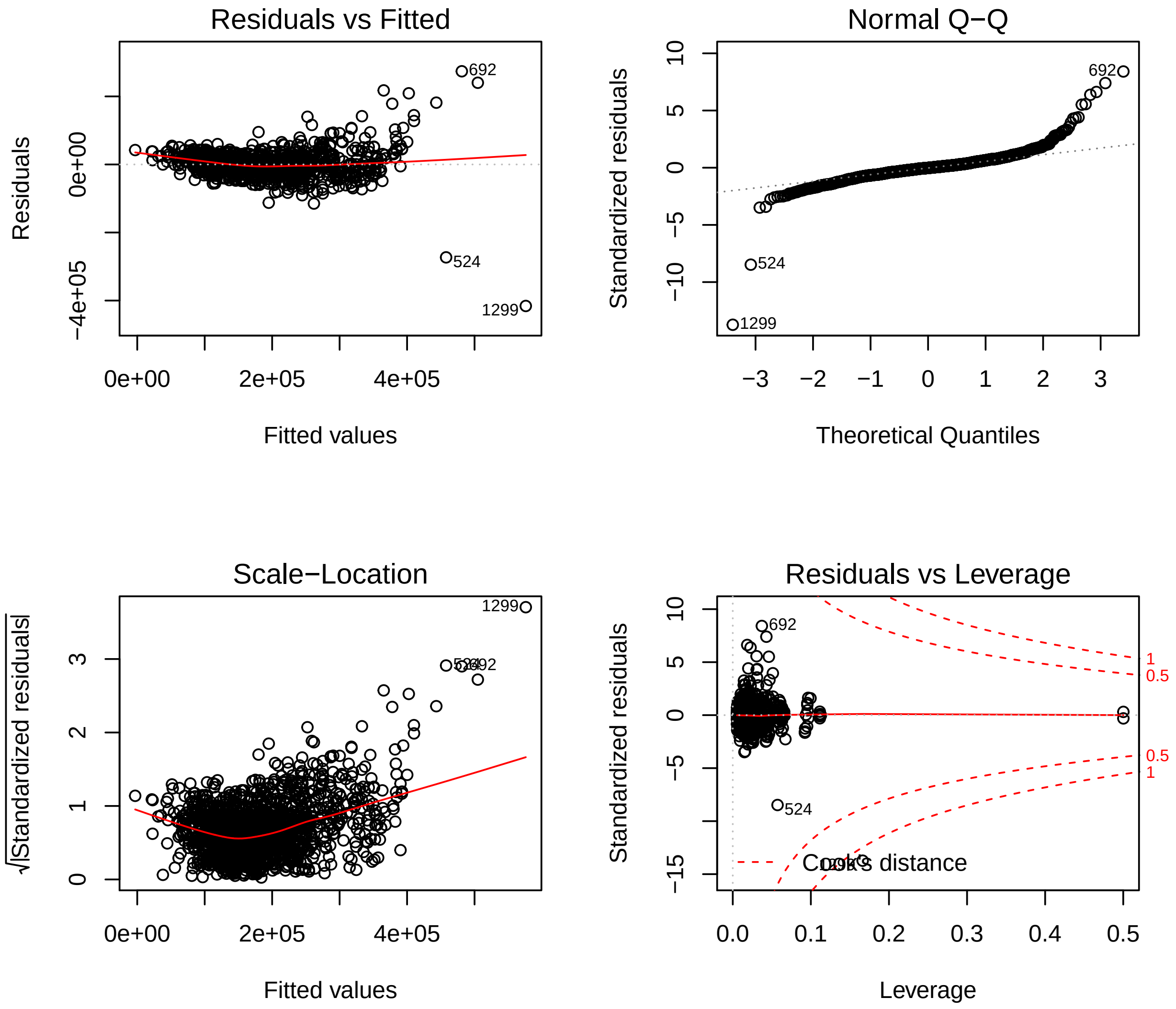
```r
fit <- lm(SalePrice ~ MSSubClass + Neighborhood +
            OverallQual + YearRemodAdd + BsmtFinSF1 +
            GrLivArea + GarageCars, data = train.df)

stargazer(fit, header = FALSE, no.space = TRUE,
          style = "all2", font.size = "footnotesize",
          single.row = TRUE, intercept.bottom = FALSE)
```

Table 7:

	Dependent variable:
	SalePrice
Constant	−621,777.700*** (111,442.300)
MSSubClass	−252.678*** (24.405)
NeighborhoodBlueste	−12,510.880 (24,882.880)
NeighborhoodBrDale	−16,417.030 (11,866.240)
NeighborhoodBrkSide	−14,640.440 (9,753.068)
NeighborhoodClearCr	7,771.633 (10,647.930)
NeighborhoodCollgCr	−8,855.995 (8,742.258)
NeighborhoodCrawfor	7,447.104 (9,729.488)
NeighborhoodEdwards	−24,629.000*** (9,317.138)
NeighborhoodGilbert	−12,303.590 (9,065.933)
NeighborhoodIDOTRR	−25,820.490** (10,379.870)
NeighborhoodMeadowV	1,205.989 (11,879.710)
NeighborhoodMitchel	−18,072.460* (9,692.696)
NeighborhoodNAmes	−18,834.730** (8,971.063)
NeighborhoodNoRidge	40,530.480*** (10,050.780)
NeighborhoodNPkVill	−9,409.847 (13,823.590)
NeighborhoodNridgHt	49,514.520*** (9,100.426)
NeighborhoodNWAmes	−21,429.510** (9,344.861)
NeighborhoodOldTown	−31,650.010*** (9,111.020)
NeighborhoodSawyer	−19,047.140** (9,502.205)
NeighborhoodSawyerW	−15,759.290* (9,404.611)
NeighborhoodSomerst	7,011.384 (8,880.182)
NeighborhoodStoneBr	55,104.420*** (10,602.380)
NeighborhoodSWISU	−27,693.620** (11,081.320)
NeighborhoodTimber	3,500.186 (9,974.428)
NeighborhoodVeenker	25,505.000* (13,027.270)
OverallQual	16,039.450*** (1,094.370)
YearRemodAdd	310.102*** (56.246)
BsmtFinSF1	23.326*** (2.113)
GrLivArea	52.970*** (2.326)
GarageCars	11,974.750*** (1,614.394)
Observations	1,460
R^2	0.829
Adjusted R^2	0.826
Residual Std. Error	33,181.540 (df = 1429)
F Statistic	231.137*** (df = 30; 1429) (p = 0.000)
Note:	*p<0.1; **p<0.05; ***p<0.01

6.4.1 Diagnostic Plots

```
p <- par(mfrow=c(2,2))
plot(fit)
```



The residual plot does not look like a shotgun pattern and we may be violating the assumption of random variance. My assumption is that the high influence points and/or outliers may be impacting the model. Therefore, we will exclude the high influence points from our final model. We will also take the log of SalePrice as we began to get negative predictions after removing the high influence points.

6.5 Final Model

```r
train.df.2 <- train.df[-c(524, 1299),]
fit <- lm(I(log(SalePrice)) ~ MSSubClass + Neighborhood +
            OverallQual + YearRemodAdd + BsmtFinSF1 +
            GrLivArea + GarageCars, data = train.df.2)
```

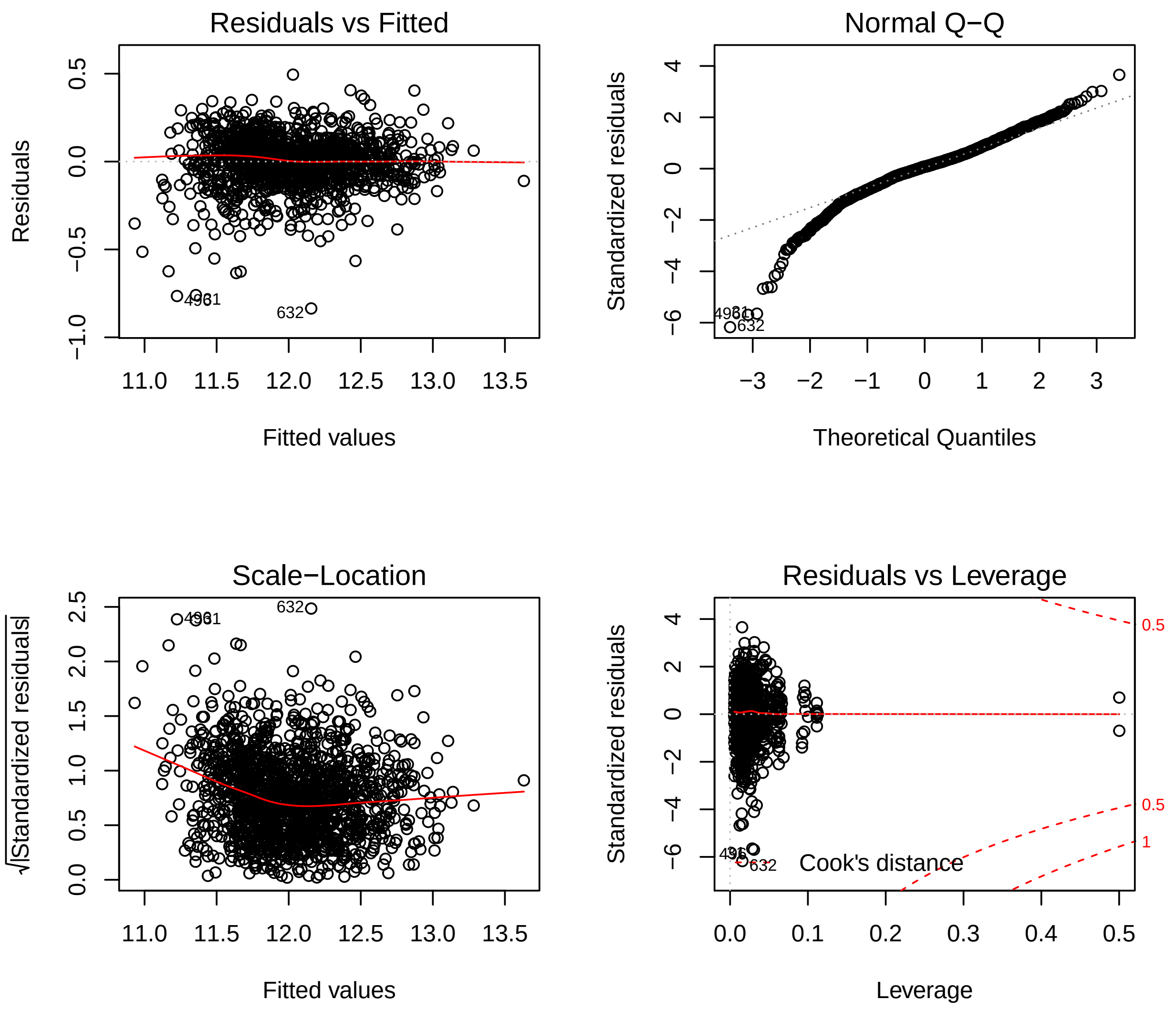
Table 8: Final Model

	Dependent variable:
	I(log(SalePrice))
Constant	5.914^{***} (0.458)
MSSubClass	-0.001^{***} (0.0001)
NeighborhoodBlueste	-0.096 (0.102)
NeighborhoodBrDale	-0.209^{***} (0.049)
NeighborhoodBrkSide	-0.110^{***} (0.040)
NeighborhoodClearCr	0.067 (0.044)
NeighborhoodCollgCr	-0.029 (0.036)
NeighborhoodCrawfor	0.046 (0.040)
NeighborhoodEdwards	-0.126^{***} (0.038)
NeighborhoodGilbert	-0.030 (0.037)
NeighborhoodIDOTRR	-0.265^{***} (0.043)
NeighborhoodMeadowV	-0.157^{***} (0.049)
NeighborhoodMitchel	-0.078^{*} (0.040)
NeighborhoodNAmes	-0.064^{*} (0.037)
NeighborhoodNoRidge	-0.012 (0.042)
NeighborhoodNPkVill	-0.061 (0.057)
NeighborhoodNridgHt	0.072^{*} (0.038)
NeighborhoodNWAmes	-0.067^{*} (0.038)
NeighborhoodOldTown	-0.212^{***} (0.037)
NeighborhoodSawyer	-0.079^{**} (0.039)
NeighborhoodSawyerW	-0.078^{**} (0.039)
NeighborhoodSomerst	0.019 (0.037)
NeighborhoodStoneBr	0.093^{**} (0.044)
NeighborhoodSWISU	-0.131^{***} (0.046)
NeighborhoodTimber	-0.004 (0.041)
NeighborhoodVeenker	0.105^{**} (0.054)
OverallQual	0.091^{***} (0.005)
YearRemodAdd	0.003^{***} (0.0002)
BsmtFinSF1	0.0002^{***} (0.00001)
GrLivArea	0.0003^{***} (0.00001)
GarageCars	0.069^{***} (0.007)
Observations	1,457
R^2	0.885
Adjusted R^2	0.882
Residual Std. Error	0.136 (df = 1426)
F Statistic	365.465^{***} (df = 30; 1426) (p = 0.000)
Note:	*p$<$0.1; **p$<$0.05; ***p$<$0.01

There are some interesting coeffecients in our final model. It is unsurprising that we see certain neighboorhoods have greater impact to the sale price but it is interesting to see the specific ones. I am also surprised that the number of cars that a garage can hold has such a great impact. Intuitively, it makes sense that more cars indicates greater wealth and a higher sale price but it was expected to see it represented in our model.

6.5.1 Diagnostic Plots

```
fit.plot <- lm(log(SalePrice) ~ MSSubClass + Neighborhood +
             OverallQual + YearRemodAdd + BsmtFinSF1 +
             GrLivArea + GarageCars, data = train.df.2)
p <- par(mfrow=c(2,2))
plot(fit.plot)
```



Now we see a more random variance in our diagnostic plots and I am more comfortable with our model. Taking the log of our outcome variable has improved our ability to validate this model.

6.5.2 Multicollinearity

```
library(car)
library(data.table)
rmfit <- setDT(as.data.frame(car::vif(fit)), keep.rownames = TRUE)[]
rmfit$Adjusted_GVIF <- (rmfit$`GVIF^(1/(2*Df))`^2)
kable(rmfit, align = c("l", "c", "c", "c", "c"))
```

rn	GVIF	Df	GVIF^(1/(2*Df))	Adjusted_GVIF
MSSubClass	1.413403	1	1.188866	1.413403
Neighborhood	5.812676	24	1.037348	1.076091
OverallQual	2.988764	1	1.728804	2.988764
YearRemodAdd	1.784746	1	1.335944	1.784746
BsmtFinSF1	1.238327	1	1.112801	1.238327
GrLivArea	1.968825	1	1.403148	1.968825
GarageCars	1.936513	1	1.391587	1.936513

Using GVIF^(1/(2*Df)) [4] in order to verify that the VIF threshold of 5 for multicollinearity is not exceed. Fortunately, we find that no variable exceeds the threshold and we do not need to adjust for multicollinearity.

6.6 Prediction results with test data set using final model

First there are some observations that are missing values included in our model. For this reason, we will use imputation to complete the cases so our predictions can be carried out.

```
test.df  <- as_tibble(read.csv(paste("https://raw.githubusercontent.com/",
                                     "ChristopheHunt/",
                                      "MSDA---Coursework/master",
                                      "/Data%20605/Final%20Project/test.csv",
                                     sep = "")))
```

6.6.1 Imputation

The test data provided by Kaggle has `NAs` for some of our independent variables. As such, we will use a non-parametic method of imputation so that we can run our final model.

```
library(missForest)
registerDoParallel(cl = makeCluster(8), cores = 8)
set.seed(1234)
test.df.imp <- as.data.frame(test.df %>% dplyr::select(Id, MSSubClass, Neighborhood,
                                                       OverallQual, OverallCond,
                                                       YearBuilt, YrSold,
                                                       YearRemodAdd, BsmtFinSF1,
                                                       GrLivArea,GarageCars)) %>%
  missForest(maxiter = 10, ntree = 100,
             replace = TRUE, parallelize = 'forests', verbose = TRUE)

write.csv(test.df.imp$ximp,"imputed_test_data.csv", row.names = FALSE)
#imputed_data to csv file due to processing time taken by missForest
```

[4] "Which Variance Inflation Factor Should I Be Using: $GVIF$ or $textGVIF^{1/(2cdottextdf)}$?" R. N.p., n.d. Web. 13 Nov. 2016.

6.6.2 Prediction results

```r
test.df  <- as_tibble(read.csv(paste("https://raw.githubusercontent.com/",
                                     "ChristopheHunt/",
                                      "MSDA---Coursework/master",
                                      "/Data%20605/Final%20Project/imputed_test_data.csv",
                                     sep = "")))

SalePrice.predict <- exp(predict.lm(fit, type = "response", newdata = test.df))
test.df.wpredict <- cbind(test.df %>% dplyr::select(Id), round(SalePrice.predict,0))
colnames(test.df.wpredict)[2] <- c("SalePrice")
kable(head(test.df.wpredict), split.table = Inf)
```

Id	SalePrice
1461	116767
1462	153700
1463	177987
1464	188130
1465	207117
1466	172621

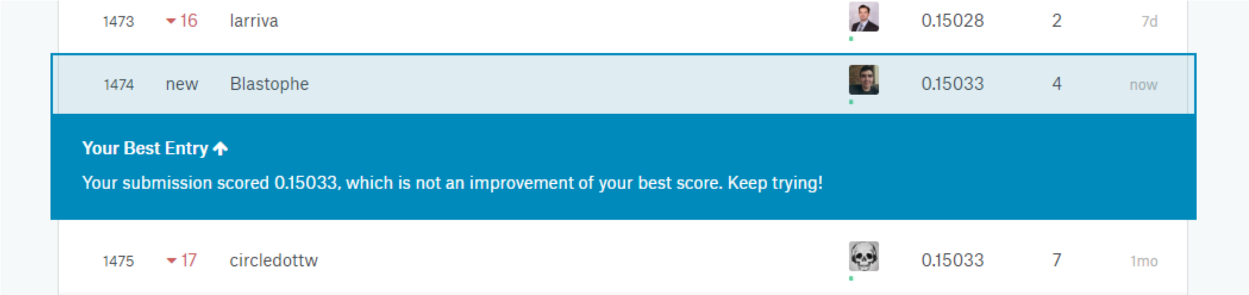
6.6.3 Prediction results for submission

```r
gz1 <- gzfile("submission.csv.gz", "w")
write.csv(test.df.wpredict, gz1, row.names = FALSE)
close(gz1)
```

6.7 Kaggle Results

Kaggle.com user name: `Blastophe`

Kaggle.com score: `0.15033`



1473 ▼16 larriva 0.15028 2 7d

1474 new Blastophe 0.15033 4 now

Your Best Entry ↑

Your submission scored 0.15033, which is not an improvement of your best score. Keep trying!

1475 ▼17 circledottw 0.15033 7 1mo

Figure 1: